Ezinne Okechukwu | Data Scientist | adahacos@gmail.com

In [1]:

```python
# import necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```python
# import dataset
df_students = pd.read_csv('StudentsPerformance.csv')
```

Data Inspection and Preparation

In [3]:

```python
# view top 5 rows
df_students.head()
```

Out[3]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

- This is a dataset of Students Performance containing their gender, race/ethnicity, parental level of education, lunch type, info on whether they completed the test preparation course and scores for the 3 courses.

In [4]:

```python
# get a statistical summary
df_students.describe()
```

Out[4]:

	math score	reading score	writing score
count	1000.00000	1000.000000	1000.000000
mean	66.08900	69.169000	68.054000
std	15.16308	14.600192	15.195657
min	0.00000	17.000000	10.000000
25%	57.00000	59.000000	57.750000
50%	66.00000	70.000000	69.000000
75%	77.00000	79.000000	79.000000
max	100.00000	100.000000	100.000000

In [5]:

```python
# summary of all columns
df_students.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 8 columns):
 #   Column                       Non-Null Count  Dtype 
---  ------                       --------------  ----- 
 0   gender                       1000 non-null   object
 1   race/ethnicity               1000 non-null   object
 2   parental level of education  1000 non-null   object
 3   lunch                        1000 non-null   object
 4   test preparation course      1000 non-null   object
 5   math score                   1000 non-null   int64 
 6   reading score                1000 non-null   int64 
 7   writing score                1000 non-null   int64 
dtypes: int64(3), object(5)
memory usage: 62.6+ KB
```

In [6]:

```python
df_students.shape
```

Out[6]:

```
(1000, 8)
```

In [7]:

```python
# check for null
df_students.isnull().sum()
```

Out[7]:

```
gender                         0
race/ethnicity                 0
parental level of education    0
lunch                          0
test preparation course        0
math score                     0
reading score                  0
writing score                  0
dtype: int64
```

- There is no null field in the dataframe.

In [8]:

```python
# check for duplicate
df_students.drop_duplicates().head()
```

Out[8]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

- No duplicate record was found.

In [9]:

```python
# let's add two more columns average score and class position
df_students['average score'] = round((df_students['math score'] + df_students['reading score']
                                      + df_students['writing score'])/3, 2)
df_students['class position'] = np.floor(df_students['average score'].rank(ascending = False)).astype(int)
df_students.head()
```

Out[9]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score	average score	class position
0	female	group B	bachelor's degree	standard	none	72	72	74	72.67	383
1	female	group C	some college	standard	completed	69	90	88	82.33	160
2	female	group B	master's degree	standard	none	90	95	93	92.67	32
3	male	group A	associate's degree	free/reduced	none	47	57	44	49.33	902
4	male	group C	some college	standard	none	76	78	75	76.33	284

- Two more columns; average score and class position were added to the dataframe. Our dataset is a clean one, so let's move on.

Explanoratory Data Analysis

In [10]:

```python
# What is the total no of students?
students = len(df_students)
print(f'Total Students: {students}\n')

# How many ethnic groups are there?
race = df_students['race/ethnicity'].nunique()
print(f'Total Race/Ethnicity: {race}\n')

# How many groups of parental levels are there?
parents = df_students['parental level of education'].nunique()
print(f'Parental Level of Education: {parents}\n')

# How many lunch groups are there?
lunch = df_students['lunch'].nunique()
print(f'Different Lunch Groups: {lunch}')
```

```
Total Students: 1000

Total Race/Ethnicity: 5

Parental Level of Education: 6

Different Lunch Groups: 2
```

Descriptive Statistics

In [11]:

```
# What is the median of the math, writing, and reading score?
print('    What is the median of the math, writing, and reading score?')
print(df_students[['math score', 'writing score', 'reading score']].median().sort_values())

# What is the mean of the math, writing, and reading score?
print('\n  What is the mean of the math, writing, and reading score?')
print(df_students[['math score', 'writing score', 'reading score']].mean().sort_values())
```

```
    What is the median of the math, writing, and reading score?
math score       66.0
writing score    69.0
reading score    70.0
dtype: float64

  What is the mean of the math, writing, and reading score?
math score       66.089
writing score    68.054
reading score    69.169
dtype: float64
```

In [12]:

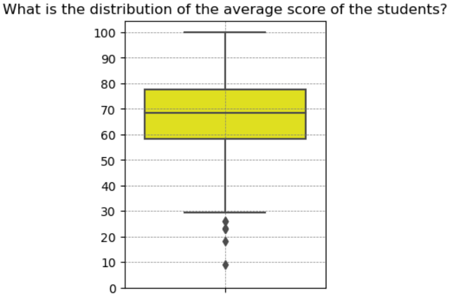
```
# What is the mode of the math, writing, and reading score?
df_students[['math score', 'writing score', 'reading score']].mode()
```

Out[12]:

	math score	writing score	reading score
0	65	74	72

In [13]:

```
# What is the distribution of the average score?
plt.figure(figsize=(3,4))
sns.boxplot(y='average score', data=df_students, color = 'yellow')
plt.ylabel('')
plt.yticks(range(0,105,10))
plt.grid(True, which = 'both', linestyle='--', linewidth=0.5, color='gray')
plt.title('What is the distribution of the average score of the students?')
plt.show()
```

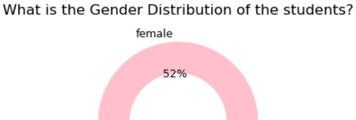


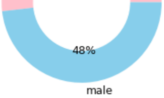
The median average score is slightly below 70, upper quartile is roughly 80 and we've like 4 outliers, these score are below 30.

Demographic Analysis

In [14]:

```
# What is the gender distribution of the students?
plt.figure(figsize=(3,3))
df_students['gender'].value_counts().plot(kind = 'pie', colors =['pink', 'skyblue'], fontsize=9, autopct='%1.
plt.title('What is the Gender Distribution of the students?', fontsize=12)
pie_radius = 1
hole_radius = 0.6
plt.gca().add_artist(plt.Circle((0,0), hole_radius, color='white'))
plt.ylabel('')
plt.show()
```

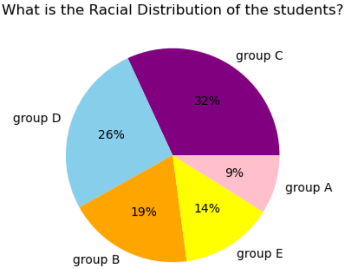




- The total no of Female students is slightly above the total no of Male students.

In [15]:

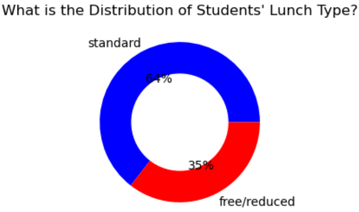
```
# What is the racial distribution of the students?
plt.figure(figsize=(6,4))
df_students['race/ethnicity'].value_counts().plot(kind = 'pie', autopct='%1.0f%%',
                                                  colors=['purple', 'skyblue', 'orange', 'yellow', 'pink'])
plt.xlabel('')
plt.ylabel('')
plt.title('What is the Racial Distribution of the students?')
plt.show()
```



- Group C is the Most Populated Race while Group A is the Least Populated Race.

In [16]:

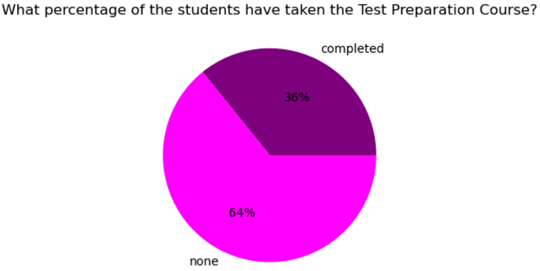
```
# What is the distribution of lunch types?
plt.figure(figsize=(3,3))
df_students['lunch'].value_counts().plot(kind = 'pie', colors =['blue', 'red'], fontsize=10, autopct='%1.0f%%
pie_radius = 1
hole_radius = 0.6
plt.gca().add_artist(plt.Circle((0,0), hole_radius, color='white'))
plt.ylabel('')
plt.title("What is the Distribution of Students' Lunch Type?", fontsize=12)
plt.show()
```



- Only 64% of the students enjoy a standard lunch.

In [17]:

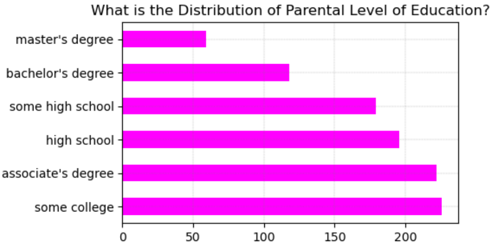
```
# how % of students have taken the test preparation course
plt.figure(figsize=(6,4))
df_students['test preparation course'].value_counts().sort_values(ascending=True).plot(kind = 'pie', autopct=
                                                  colors=['purple', 'magenta'])
plt.xlabel('')
plt.ylabel('')
plt.title('What percentage of the students have taken the Test Preparation Course?')
plt.show()
```



- Only 36% of the students found it necessary to complete the test preparation course.

In [18]:

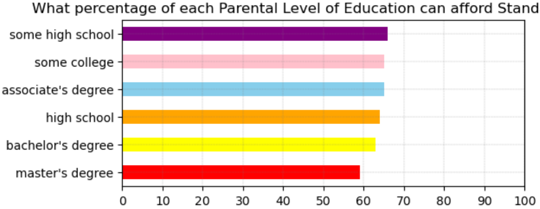
```
# group students by parental level of education
plt.figure(figsize=(5,3))
x = df_students['parental level of education'].value_counts().sort_values(ascending=False)
x.plot(kind = 'barh', color = 'magenta')
plt.ylabel('')
plt.grid(True, which='both', linewidth=0.2, linestyle="--", color = 'gray')
plt.title("What is the Distribution of Parental Level of Education?", fontsize=12)
plt.show()
```



- Only a few of the parents are privileged to have a Masters Degree. Most went to 'some college' or have an associate degree. Let's look at what percentage of each group can afford standard lunch for their kids.

In [19]:

```
grouped = df_students.groupby(['parental level of education', 'lunch']).size().unstack()
grouped['total'] = grouped.sum(axis = 1)
grouped['standard_perc'] = round((grouped['standard']/grouped['total'] * 100), 0)
grouped = grouped.sort_values('standard_perc')

# visualize percentage of parental group who can afford standard meal
plt.figure(figsize=(6,2.5))
grouped['standard_perc'].plot(kind='barh', color=['red', 'yellow', 'orange', 'skyblue', 'pink','purple'])
plt.ylabel('')
plt.grid(True, which = 'both', linestyle = '--', linewidth = 0.2, color = 'gray')
plt.xticks([0,10,20,30,40,50,60,70,80,90,100])
plt.title('What percentage of each Parental Level of Education can afford Standard lunch?')
plt.show()
```

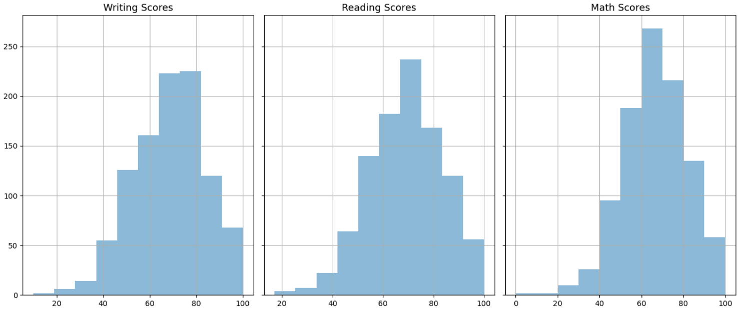


It is surprising to see that students whose parents have low level of education ensure their kids have a standard lunch while students whose parents have masters degree are not very concerned about their children's nutrition.

Distribution Analysis

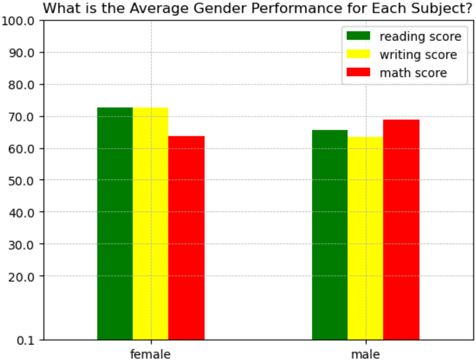
In [20]:

```
# distribution of math, writing and reading score

fig, (ax1, ax2, ax3) = plt.subplots(1,3, sharey = True, figsize = (14,6))
df_students['writing score'].hist(ax=ax1, bins = 10, alpha = 0.5)
ax1.set_title('Writing Scores', fontsize=13)

df_students['reading score'].hist(ax=ax2, bins = 10, alpha = 0.5)
ax2.set_title('Reading Scores', fontsize=13)

df_students['math score'].hist(ax=ax3, bins = 10, alpha = 0.5)
ax3.set_title('Math Scores', fontsize=13)

plt.tight_layout()
plt.show()
```



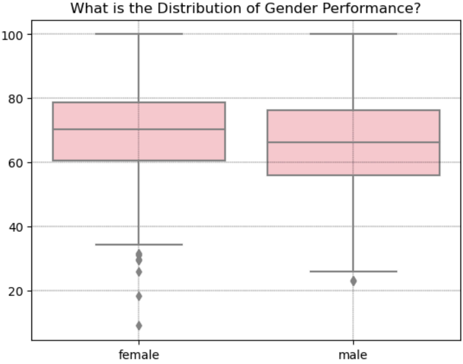
In [21]:

```
# gender performance for the 3 subjects
x = df_students.groupby('gender')[['reading score', 'writing score', 'math score']].mean().reset_index()
x.plot(kind = 'bar', x='gender', y=['reading score', 'writing score', 'math score'], color = ['green', 'yello
plt.xticks(rotation = 360)
plt.xlabel('')
plt.yticks([0.10,20,30,40,50,60,70,80,90,100])
plt.legend(bbox_to_anchor=(1,1))
plt.grid(True, which ='both', linestyle = '--', linewidth=0.5)
plt.title('What is the Average Gender Performance for Each Subject?')
plt.show()
```



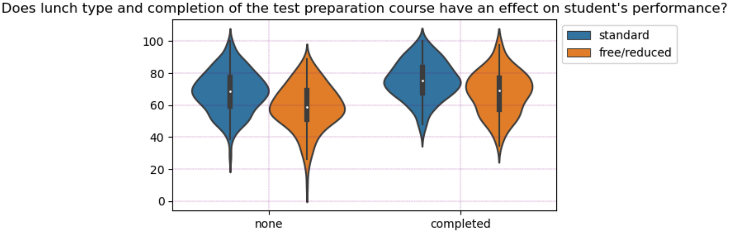
- Females perform better than the males in all courses except for math. Let's see the distribution of gender scores.

In [22]:
```
sns.boxplot(x='gender', y='average score', data=df_students, color = 'pink')
plt.xlabel('')
plt.ylabel('')
plt.title('What is the Distribution of Gender Performance?')
plt.xticks(rotation = 360)
plt.grid(True, which = 'both', linestyle = '--', linewidth=0.3, color = 'black')
plt.show()
```



Impact of Lunch Type and Test Preparation Course

In [23]:
```
# Does lunch type and test preparaion have an effect on student's performance?
plt.figure(figsize=(6,3))
sns.violinplot(x='test preparation course', y = 'average score', data = df_students, hue = 'lunch')
plt.xlabel('')
plt.ylabel('')
plt.grid(True, which = 'both', linestyle = "--", linewidth=0.2, color = 'purple')
plt.title("Does lunch type and completion of the test preparation course have an effect on student's performa
plt.legend(bbox_to_anchor = (1,1))
plt.show()
```

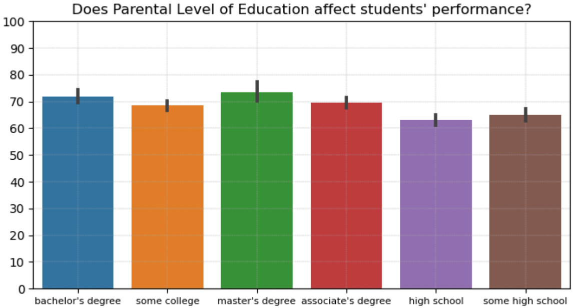


As could have been predicted, students with free/reduced lunch who also did not complete the test preparation course have the worst performance. Students with standard lunch who completed their test preparation course have the best scores.

Impact of Parental Level of Education

In [24]:

```
plt.figure(figsize = (8,4))
sns.barplot(x='parental level of education', y = 'average score', data=df_students)
plt.xlabel('')
plt.ylabel('')
plt.xticks(fontsize=8)
plt.yticks([0,10,20,30,40,50,60,70,80,90,100])
plt.grid(True, which = 'both', linestyle = '--', linewidth=0.3)
plt.title("Does Parental Level of Education affect students' performance?")
plt.show()
```

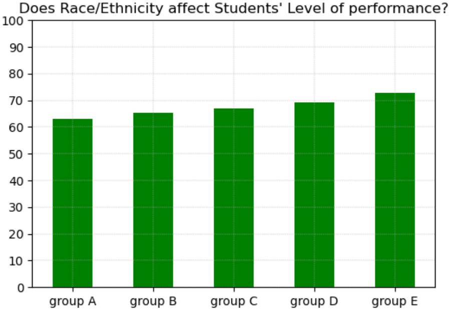


Students performance is slightly influenced by their parents level of education. Parents who are more educated tend to have more brilliant students than those whose parents went to some high school or a high school.

Impact of Ethnicity on Students' Performance

In [25]:

```
# Does Ethnicity affect Students' Level of performance?
plt.figure(figsize=(6,4))
df_students.groupby('race/ethnicity')['average score'].mean().plot(kind = 'bar', color = 'green')
plt.xlabel('')
plt.ylabel('')
plt.title("Does Race/Ethnicity affect Students' Level of performance?")
plt.xticks(rotation = 360)
plt.yticks([0,10,20,30,40,50,60,70,80,90,100])
plt.grid(True, which = 'both', linestyle ="--", linewidth = 0.3)
plt.show()
```



- Group E ethnicity has the best academic performance.

In [26]:

```
# For each ethnic group what is their best performing subjects?
x = df_students.groupby('race/ethnicity')[['math score', 'writing score', 'reading score']].mean().astype('in
x.style.background_gradient(subset=['math score', 'writing score', 'reading score'], cmap='YlOrRd_r')
```

Out[26]:

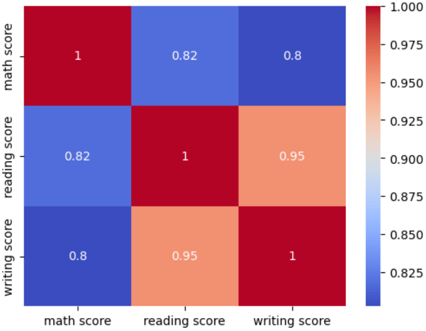
race/ethnicity	math score	writing score	reading score
group A	61	62	64
group B	63	65	67
group C	64	67	69
group D	67	70	70
group E	73	71	73

- Group E has the highest score for all three subjects. While other ethnicities are lagging behind in Mathematics, Group E had their best score in Math. Other ethnicities had their best score in reading or writing but Ethnic E had their pooorest score in writing. Other ethnicities seem to be terrified with math. Group A had the poorest performance for all three subjects.

Correlative Analysis

In [27]:

```
correlation_matrix = df_students[['math score', 'reading score', 'writing score']].corr()
plt.figure(figsize=(6,4.5))
sns.heatmap(correlation_matrix, annot = True, cmap = 'coolwarm')
plt.show()
```



- Writing and reading score has a strong positive linear relationship compared to 'math score vs reading score' and 'math score vs writing score'.

Top 10 & Worst 10 Performing Students

In [28]:

```
# Let's look at 10 top performing students and their background
print('What is the Demographics of Top 10 Performing Students?')
df_students[['gender', 'race/ethnicity', 'parental level of education',
             'lunch', 'average score', 'class position']].sort_values(by = ['average score', 'gender'],
                                                                       ascending = [False, True]).head(10)
```

```
What is the Demographics of Top 10 Performing Students?
```

Out[28]:

	gender	race/ethnicity	parental level of education	lunch	average score	class position
458	female	group E	bachelor's degree	standard	100.00	2
962	female	group E	associate's degree	standard	100.00	2
916	male	group E	bachelor's degree	standard	100.00	2
114	female	group E	bachelor's degree	standard	99.67	4
179	female	group D	some high school	standard	99.00	5
712	female	group D	some college	standard	99.00	5
165	female	group C	bachelor's degree	standard	98.67	7
625	male	group D	some college	standard	98.67	7
685	female	group E	master's degree	standard	97.67	10
903	female	group D	bachelor's degree	free/reduced	97.67	10

90% of students in the top 10 have a standard lunch, 80% are females, majority of them are in group E and group D ethnicity and 50% of their parents have a bachelor's degree.

In [29]:

```
# Demographic of 10 worst performing students and their background
print('What is the demographic of Least 10 Performing Students?')
df_students[['gender', 'race/ethnicity', 'parental level of education',
             'lunch', 'average score', 'class position']].sort_values(by = ['average score', 'gender'],
                                                                       ascending = [False, True]).tail(10)
```

```
What is the demographic of Least 10 Performing Students?
```

Out[29]:

	gender	race/ethnicity	parental level of education	lunch	average score	class position
211	male	group C	some college	free/reduced	30.00	991
338	female	group B	some high school	free/reduced	29.67	992
787	female	group B	some college	standard	29.67	992
601	female	group C	high school	standard	29.33	994
17	female	group B	some high school	free/reduced	26.00	995
76	male	group E	some high school	standard	26.00	995
327	male	group A	some college	free/reduced	23.33	997
596	male	group B	high school	free/reduced	23.00	998
980	female	group B	high school	free/reduced	18.33	999
59	female	group C	some high school	free/reduced	9.00	1000

70% of the worst performing students have free/reduced lunch, 40% of their parents went to some high school, 30% went to high school and the other 30% went to some college, 60% are females and no group D student is amongst the last 10.

Conclusion

- Nutrition affects academic performance seeing that students who had standard lunch perform better than students with reduced/free lunch.
- The more educated a parent is, the more likely it is for the offspring to excel in academics.
- The females are leading the males in average overall performance and all courses except for math.
- Ethnic group E significantly outperforms the other races in all three courses.
- Students who complete the test preparation course are more likely to perform better than students who do not.
- The average student find reading to be easier than maths judging from average scores.
- 70 of Students in the worst performing 10 have a free/reduced lunch while 90% of 10 best perfoming students have a standard lunch.